COMUNICACIÓN EN TIEMPO REAL SOBRE INTERNET

Felipe Moreno Strauch

Estudiante de la ETSETB, UPC y socio colaborador de BJT.
felipe@bjt.upc.es

INTRODUCCIÓN

En los primeros 20 años de su existencia, Internet era básicamente utilizada para el intercambio de mensajes de correo electrónico y para la transferencia de ficheros y casi exclusivamente por personal técnico o de investigación de las universidades, de instituciones del gobierno o laboratorios de investigación de la Industria. Pero en los últimos años, con el crecimiento exponencial, la aparición de nuevos servicios y una transición hacia una red comercial la situación ha cambiado radicalmente.

El significativo aumento del ancho de banda y de la capacidad de proceso de los ordenadores son dos factores que, juntamente con la evolución de las tecnologías de acceso nos ha permitido avanzar hacia el concepto de Integración de Servicios. Desarrollar una sola red, capaz de soportar todos los servicios que actualmente viajan por distintas redes de transporte: la difusión de televisión y radio, la telefonía, la transmisión de ficheros, etc. Pero no solo esto, sino además, asegurar para cada uno de ellos, la calidad de servicio requerida, lo que supone ser capaz de tratar los distintos tipos de tráfico generado de forma óptima.

Empiezan a aparecer nuevos servicios que funcionarán en Internet y que van mas allá que la simple distribución de páginas web o envío de mensajes de correo electrónico. Nuevos servicios como la difusión de vídeo y audio (tanto desde ficheros como retransmisión en directo de eventos), Radios y Televisión vía web, vídeo bajo demanda, telefonía, videoconferencia, presentaciones multimedia, simulaciones en tiempo real, etc.

Aunque por ahora algunos aún no se pueden implementar en la practica debido a la escasez de ancho de banda, otros ya empiezan a ponerse en marcha y los resultados son muy interesantes. La industria se ha interesado: Microsoft ha creado una nueva división para desarrollar aplicaciones multimedia (Windows Media Player) y ha entrado de lleno en la batalla con Real Networks (RealAudio y RealVideo) por convertirse en el standard para la transmisión multimedia en tiempo real. Las grandes empresas del sector de las comunicaciones como la NBC o CNN ven Internet como un mercado potencial muy interesante y ya han empezado a explorarlo (en 1997 60.000 usuarios siguieron en directo por Internet un capitulo de la serie ER). La posibilidad de incluir publicidad es un atractivo añadido ya que se puede enviar los anuncios en función del perfil de cada usuario.

Pero para que todo ello pueda realmente evolucionar es necesario estudiar la transmisión de información en tiempo real sobre una red de conmutación de paquetes como Internet, preparada para transmitir datos.

Hay que buscar una forma de adaptar este nuevo tráfico, que tiene unas características muy singulares a la infraestructura disponible, estudiando los posibles problemas y desarrollando nuevos protocolos.

TRANSMISIÓN EN TIEMPO REAL SOBRE REDES DE CONMUTACIÓN DE PAQUETES

La comunicación en tiempo real tiene como característica importante el hecho de que el valor de la comunicación depende del momento en que los mensajes llegan al destino. Al contrario de lo que ocurre con la transmisión de datos a las que estamos acostumbrados (por ejemplo una transmisión de fichero), los paquetes que integran una transmisión de vídeo o audio deben llegar al destino en el momento adecuado, o como mucho dentro de un cierto margen ya que el sistema los necesita en aquel momento para reproducir la señal. Existe una cuota para el retardo con que un mensaje se ha de entregar correctamente al destino. Este retardo máximo aceptable fija una especie de "fecha de caducidad" para la información o tiempo limite para la llegada del mensaje. Si un mensaje llega una vez ya ha caducado, es decir sobrepasa el tiempo límite, el valor de la información que contiene disminuye y muchas veces acaba siendo completamente inútil (como una muestra de señal de audio que llegue con un retraso de 2 segundos) y acaban por ser descartados y se consideran, desde el punto de vista de la conexión, como paquetes perdidos.

En función de la tolerancia a que los paquetes lleguen con un retardo mayor que el aceptable se puede clasificar la comunicación en tiempo real como "hard real-time" o "soft real-time". En el primer caso el servicio no es capaz de tolerar perdidas (por ejemplo un sistema de control remoto, cuando ha de reaccionar frente a una emergencia), mientras que en el segundo es aceptable una cierta tasa de perdida (por ejemplo un sistema de transmisión de audio).

El retardo acumulado desde un extremo a otro de la comunicación se llama latencia. A la latencia contribuyen el origen, la red y el destino. El origen contribuye con el tiempo que transcurre desde que muestrea la señal hasta que envía las muestras y el destino con el tiempo que tarda antes de analizarlo. Estos retardos en general no son significativos frente a los que impone la red. La red contribuye de diferentes formas al retardo total:

- Retardo de propagación: es el tiempo que tarda la información en viajar desde un extremo a otro sobre el medio de transmisión utilizado. En entorno reducidos este retardo es siempre despreciable, pero al hablar de sistemas que funcionen sobre Internet, a escala global se ha de tener en cuenta. Por ejemplo, considerando la velocidad de la luz como velocidad de propagación una señal tardaría unos 134 ms en dar la vuelta a la Tierra sobre el ecuador. Teniendo en cuenta que las restricciones impuestas rondan las centenas de ms este factor puede ser importante.

- Retardo de transmisión: es el tiempo en que el origen tarda en poner el paquete en el medio de transmisión. Viene determinado por la velocidad de transmisión y por el tamaño del paquete.

- Retardo "store-and-forward": es el tiempo que se pierde debido a que los routers intermedios deben recibir el paquete completamente antes de retransmitirlo. Depende del número de nodos que atraviesa el paquete.

- Retardo de proceso: debido a que los routers deben analizar la cabecera y decidir la ruta que debe seguir. Además es necesario cambiar algunos campos y rehacer cálculos de checksum (por ejemplo al cambiar el campo time-to-live de un paquete IP).

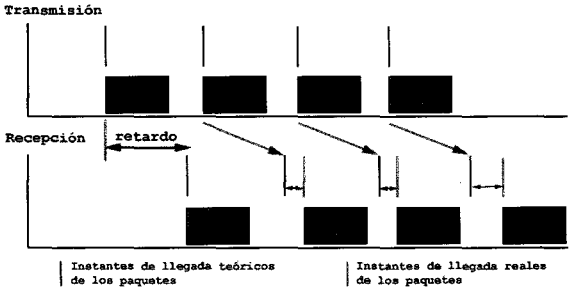

Figura 1. *Efecto de la transmisión de un flujo constante de información sobre una red de conmutación de paquetes.*

Otro parámetro que hemos de tener en cuenta al analizar los servicios que requieren tiempo real es el jitter, que se podría definir como la máxima variación del retardo extremo a extremo que sufren los paquetes de una misma conexión. Como acabamos de ver, muchos factores contribuyen al retardo extremo a extremo, básicamente retardos introducidos por la red. Lógicamente estos retardos no son deterministas y dependen del estado actual de la red lo que hace que los distintos paquetes lleguen al destino con diferentes retardos o lo que es lo mismo, que el tiempo entre la llegada de paquetes consecutivos será una variable aleatoria. Esto causará una "perdida de sincronismo" en el receptor que debe ser corregida para que sea posible reconstruir el flujo de datos original.

Este efecto es especialmente malo, por ejemplo, en transmisiones de audio ya que produce una serie de chasquidos que resultan muy molestos en la comunicación. La solución que permite corregirlo es sencilla y se basa en la utilización de un buffer en recepción (con política FIFO) que se suele denominar "playout buffer". En lugar de procesar lo paquetes a medida en que llegan, el receptor los almacena en el buffer y cuando este se ha llenado los empieza a procesar. Como en media la tasa de paquetes que entra es igual a la que sale del buffer el sistema es estable.

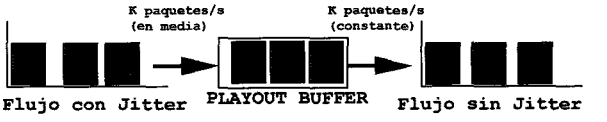

Figura 2. *Utilización del Playout Buffer para corregir el*

La utilización del playout buffer reduce el efecto del jitter de la misma forma que el sistema de amortiguación de un coche reduce el efecto de las irregularidades de la carretera. Las variaciones que son menores que el tamaño total del buffer no son observables a la salida. El cálculo del tamaño que debe tener el buffer se ha de hacer basándose en una estimación de la estadística del retardo que introduce la red y la tasa de perdidas que se acepta como tolerable y se hace mediante el percentil.

Hay que observar sin embargo que para compensar el jitter de esta forma estamos aumentando la latencia ya que ahora los paquetes han de estar almacenados en el buffer durante un tiempo antes de ser procesados. Aunque en general siempre será necesario controlar el jitter, la latencia sólo es importante en aquellos servicios que requieren interacción entre las partes implicadas. Por ejemplo para telefonía la tolerancia en el retardo total en ir y volver (round-trip delay) es de 400 ms (un retardo mayor degradaría la calidad por encima de lo aceptable) mientras que en el caso de la difusión de vídeo podríamos hablar de 500 ms o más ya que la información circula solo en un sentido. El problema en las aplicaciones que requieren interactividad es que el tiempo entre que el usuario

actúa y percibe el resultado de su actuación es un factor que influye mucho en la calidad percibida. En cambio en un sistema sin interactividad, el hecho de que haya una diferencia entre el momento en que se envía una muestra de señal y el receptor la recibe no es detectable por el usuario.

Solucionar el problema de la latencia no es tan sencillo ya que hemos de buscar mecanismos que permitan reducir el retardo que sufre un paquete en la red. Alternativas como la asignación de prioridades o la reserva de recursos podrían contribuir a ello. Desde el punto de vista del procesado de señal el desarrollo de mejores sistemas de codificación y compresión de la información (reduciendo la tasa en bits por segundo que es necesario enviar) también contribuiría.

Lo primero que hemos de hacer basándonos en estos requerimientos es verificar si las herramientas que disponemos actualmente son suficientes o es necesario algo más.

PORQUÉ TCP O UDP NO SON SUFICIENTES

Aunque podría ser posible utilizar el TCP para transportar los datos de una transmisión en tiempo real hay una serie de motivos que no aconsejan su utilización:

- TCP es un protocolo que realiza comprobación de errores y retransmisión de paquetes. El sistema de retransmisión de TCP se basa en esperar durante un cierto tiempo (el time-out) la llegada de una confirmación de que el destinatario ha recibido correctamente los datos. Si esta confirmación no llega dentro del tiempo se retransmite el paquete. El problema es que cuando el emisor se da cuenta de que debe retransmitir suele ser demasiado tarde, la información ya ha perdido su valor. Para los requerimientos de tiempo real un sistema de retransmisiones resulta inútil dadas las fuertes restricciones temporales.

- TCP utiliza un control de congestión que decrementa el tamaño de la ventana de transmisión cuando hay perdidas de paquetes para evitar sobrecargar a la red. Sin embrago en transmisiones en tiempo real hay una tasa de transmisión (la que genera la fuente) que debe llegar siempre al destino y no se puede recortar.

- TCP no dispone de un sistema de distribución multicast lo que es una seria restricción en los servicios multipunto a gran escala: enviar un paquete por cada destinatario representaría una utilización ineficaz del ancho de banda.

Como alternativa podríamos pensar en UDP ya que este protocolo no utiliza ningún sistema de control de errores y permite utilizar IP multicast pero tampoco resulta adecuado ya que es demasiado sencillo y no contiene toda la información necesaria: momento de generación de los datos (necesario para reordenar muestras y recuperar el sincronismo con otros flujos de datos), información sobre la codificación utilizada, etc.

Como hemos visto, ni TCP ni UDP resultan ser útiles para el transporte de datos con requerimientos de tiempo real. Si la infraestructura de transporte disponible es insuficiente hemos de desarrollar nuevas herramientas. Necesitamos un nuevo protocolo que complete la funcionalidad de UDP.

REAL TIME TRANSPORT PROTOCOL (RTP)

La IETF (Internet Engineering Task Force) ha desarrollado un nuevo protocolo especialmente pensado para el transporte de este tipo de información, el Real Time Transport Protocol (RTP). Este nuevo protocolo incluye una serie de funciones que facilitan esta tarea: identificación del tipo de información (tipo de codificación), números de secuencia, sincronismo (timestamp) y monitorización de la calidad de servicio. Sin embargo, RTP no posee ninguna función que garantice la entrega de los paquetes dentro del periodo adecuado ni ningún otro tipo de garantía de la cualidad de servicio. Conviene resaltar también que aunque sea un protocolo de "transporte", porque regula el envío de datos de un extremo a otro, este protocolo funciona realmente en el nivel de aplicación.

RTP prácticamente no hace ninguna suposición sobre el nivel inferior sobre el que funciona excepto que el protocolo en cuestión defina el tamaño de los paquetes ("framing") ya que el RTP no incluye en su cabecera ningún campo de longitud de los datos. No se asume la existencia de ningún tipo de control de errores, ni se asume la existencia de una conexión, ni ningún mecanismo de reordenación de paquetes. En general, en una arquitectura TCP/IP el RTP se utiliza sobre UDP, aunque el protocolo es lo suficientemente general como para utilizarse en otros tipos de red, como ATM por ejemplo.

El protocolo RTP consta de dos partes muy relacionadas: la parte que transportan los datos (a la que en general se refiere como RTP) y la parte de control, referida como RTCP (Real Time Transport Control Protocol) que realiza las funciones de monitorización de la calidad de servicio y control de los participantes de una determinada sesión. El control realizado no es estricto, es decir, no se verifica si los usuarios tienen o no derecho a participar. Esta función podría ser realizada por algún otro protocolo de control de sesión como SIP (Session Initiation Protocol) por ejemplo. Cuando RTP se utiliza sobre UDP se necesitan dos puertos: uno para la transmisión de información útil y otro para la transmisión de información de control.

FORMATO DE LA CABECERA DE RTP: TRANSPORTE DE LA INFORMACIÓN ÚTIL

El paquete RTP que transporta la información incluye la cabecera básica de RTP que consta de 96 bits (12 bytes), una lista de identificadores de fuentes (CSRC, solo utilizada cuando se agregan varios flujos de datos en uno solo) y los datos. Según el tipo de datos se puede añadir una cabecera adicional con mas información, cuyo formato esta especificado en documentos diferentes para cada tipo y publicados como RFCs.

Los campos más importantes en la cabecera son:

- Payload Type (7 bits): indica el tipo de datos que lleva el paquete. Es lo que servirá a la aplicación que recibe los datos para saber como debe interpretarlos (que CODEC utilizar).

- Timestamp (32 bits): representa el instante de generación del primer octeto del campo de datos. Este instante se deriva de un reloj (reloj en el sentido de contador) que se incrementa monotonicamente y de forma lineal con el tiempo para permitir la sincronización y la estimación del jitter. La resolución de este reloj debe ser suficiente para la precisión que se quiera conseguir en la sincronización y en el calculo del jitter. Su frecuencia depende del tipo de datos transportado y se indica en la especificación del tipo en cuestión. Por ejemplo para la transmisión de audio, el reloj de timestamp se incrementa en uno por cada muestra de señal. Si se envían 160 muestras por paquete el timestamp se incrementaría en 160. El valor inicial se elige aleatoriamente. Es posible que paquetes consecutivos tengan el mismo timestamp si fueron generados en el mismo instante de tiempo (por ejemplo información de un mismo frame en un transmisión de vídeo). También es posible que paquetes consecutivos lleven timestamps que no sean monotónicos ya que es posible que, según el formato de codificación utilizado, la información no se transmita en el mismo orden en que fue muestreada.

- Sequence Number (16 bits): indica el numero de secuencia del paquete dentro del flujo de datos. El valor inicial es aleatorio y se incrementa en uno para cada paquete enviado. Se utiliza para detectar perdidas de paquetes y reordenar paquetes con el mismo timestamp (no se utilizan para ordenación de las muestras por lo comentado en el punto anterior sobre el envío en un orden diferente al de muestreo).

- SSRC (32 bits): especifica un identificador para la fuente del flujo de datos. Todos los paquetes identificados con el mismo SSRC forman parte de un mismo espacio de números de secuencia y de timestamp. El receptor agrupa los paquetes por el SSRC cuanto recibe información de mas de una fuente diferente a la vez. El valor de este campo, que ha de ser único entre todos los participantes de una misma sesión, se elige aleatoriamente. Si un mismo host genera múltiplos flujos de datos en la misma sesión (una sesión multimedia por ejemplo, con audio y vídeo) necesita un identificador diferente para cada uno.

EL PROTOCOLO DE CONTROL: RTCP

El protocolo de control RTCP consiste en una serie de paquetes, cada uno con un formato y función especificado que se envían periódicamente los participantes en una sesión para transmitir información sobre la calidad de la recepción. Si la sesión es multipunto (funciona sobre IP multicast) los paquetes se envían a la misma dirección multicast utilizada para enviar los paquetes de datos. De esta forma el envío de los paquetes RTCP sirve además como indicador de actividad para los participantes aunque estos sean miembros pasivos (solo reciben y no emiten).

Las cuatro funciones básicas que realiza la parte de control son:

- Monitorizar la calidad de servicio: la principal función del RTCP es informar al emisor sobre la calidad de la información recibida. Como esta información se envía a todos los participantes es posible detectar si los problemas generados son locales o globales.

- Relacionar cada fuente de datos con un identificador persistente (ya que el SSRC es diferente para cada flujo de datos) llamado CNAME (Canonical Name), que se utiliza para identificar el host durante una sesión y sirve para relacionar distintos flujos de datos generados por la misma fuente. El CNAME utiliza un formato del tipo: usuario@host.

- Permitir que todos sepan el numero total de participantes en una sesión. Esto es importante ya que para que el sistema no sature la red con paquetes de control cuando el numero de participantes crece, la tasa de generación de paquetes de control debe disminuir a medida que aumente el numero de usuarios para que entre todos ocupen siempre un ancho de banda fijo. Para que cada uno pueda calcular la tasa con que debe emitir es necesario estimar el numero total de participantes.

- Distribuir información necesaria para que se puedan sincronizar los distintos flujos de datos en una sesión multimedia (por ejemplo sincronizar el audio y vídeo). Cada fuente indica la relación entre su timestamp y su reloj global (wallclock), que representa el tiempo real, para cada flujo de datos que envía. Si el sincronismo de flujos se debe lograr entre fuentes situadas en maquinas diferentes los relojes de ambas máquinas deben estar sincronizados.

En la figura 3 se puede ver un esquema que resume los identificadores utilizados por un host dentro de una sesión multimedia: el CNAME que identifica el host en la sesión, los identificadores de fuente SSRC que identifican cada flujo de datos generado y los puertos RTP y RTCP para cada flujo de datos.

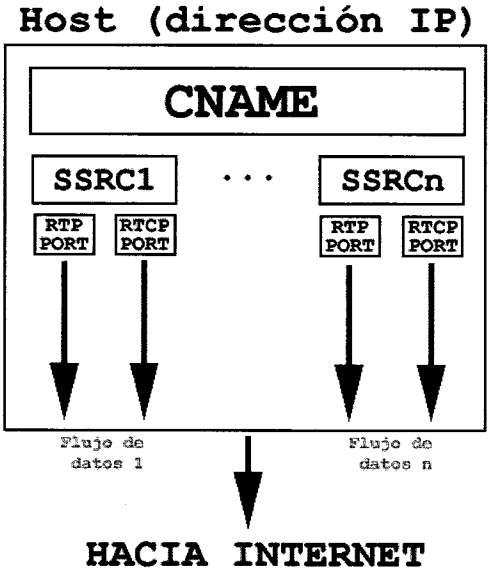

Figura 3. Esquema de identificadores de una fuente con varios flujos de datos en una sesión multimedia.

La información sobre la calidad de la recepción que se envía indica, entre otros parámetros:

- La fracción de paquetes de datos perdidos sobre el total de paquetes enviados desde el último informe.
- El número total de paquetes perdidos desde el inicio de la comunicación.
- El número de secuencia del ultimo paquete recibido correctamente.
- Una estimación del jitter, medido en las unidades del timestamp.

Esta información es enviada por todos los participantes que han recibido datos recientemente y se envía a todos los participantes (hacia la dirección del otro componente en el caso punto a punto o hacia la dirección multicast en el caso de multipunto). Se genera un informe por cada una de las fuentes de las que se ha recibido datos. En función de estos parámetros enviados el emisor puede detectar si existen problemas, si estos son locales o globales y actuar en consecuencia, por ejemplo cambiando el tipo de codificador utilizador para disminuir la tasa de información que envía a cambio de perder calidad.

CONCLUSIONES

Aunque tengamos especificadas algunas herramientas para tratar el tráfico con características de tiempo real y ya existan sistemas que funcionen actualmente en Internet, todavía nos quedan problemas que solucionar. Problemas como la garantía de la calidad de servicio (se esta desarrollando una solución basada en el protocolo RSVP – Resource Reservation Protocol – que permitirá reservar ancho de banda en los routers) o como integrar el contenido Multimedia en las paginas web (se esta desarrollando un nuevo lenguaje basado en XTML llamado SMIL – en desarrollo por el W3C – que permite definir distintos tipos de datos y una relación temporal entre ellos.

Los sistemas que funcionan actualmente en Internet son básicamente dos (se indican entre paréntesis el nombre del software cliente y el servidor): el RealSystem G2 (RealPlayer + RealServer) de Real Networks, que prácticamente fue un standard de-facto desde la creación de RealAudio en 1995 y el Windows Media (Windows Media Player + Windows Media Services) de Microsoft, creado hace tan solo un año y medio pero que ya ha empezado a ganar terreno.

La ultima versión de RealSystem utiliza el RTP como protocolo de transporte de datos y el RTSP como protocolo de control, en sustitución al RTCP. RTSP o Real Time Streaming Protocol ofrece mas opciones a la hora de monitorizar la calidad de servicio.

BIBLIOGRAFÍA

[1] Schulzrinne, H., Casner, S., Frederick, R., Jacobson, V., "RTP: A Transport Protocol for Real-Time Applications", RFC 1889, January 1996.

[2] Schulzrinne, H., "Internet Services: from Electronic Mail to Real-Time Multimedia", KIVS'95, February 1995.

[3] Aras, Ç., Kurose, J., Reeves, D., Schulzrinne, H., "Real-Time Communication in Packet-Switched Networks".

[4] IETF Audio/Video Transport (avt) charter, http://www.ietf.org/html.charters/avt-charter.html

[5] RTP: About RTP and the Audio-Video Transport Working Group, http://www.cs.columbia.edu/~hgs/rtp/

[6] Windows Media Technology en microsoft.com, http://www.microsoft.com/windows/windowsmedia/

[7] RealNetworks Documentation Library, http://service.real.com/help/library/